AUTOMATIC TRAFFIC CONTROL SYSTEM

Jemil P. Desai
Computer Science & Engineering
RNG Patel Institute of Technology
Bardoli, India
jemildesai74@gmail.com

Purvagiri V. Goswami
Computer Science & Engineering
RNG Patel Institute of Technology
Bardoli, India
cse.220840131031@gmail.com

Param N. Desai
Computer Science and Engineering
RNG Patel Institute of Technology
Bardoli, India
cse.220840131024@gmail.com

Aatman V. Barvaliya
Computer Science and Engineering
RNG Patel Institute of Technology
Bardoli, India
cse.220840131004@gmail.com

***Abstract* — Traffic congestion is a major issue in many metropolitan areas, causing significant delays and frustration for commuters. Traditional traffic light systems operate on a fixed-time basis, which fails to adapt to the real-time traffic density on different roads. This often results in inefficient traffic management, especially during peak hours or in the presence of emergency or VIP vehicles. To address this issue, we propose a density-based automatic traffic control system that dynamically adjusts signal timing based on vehicle density detected at intersections. The system is implemented using ultrasonic sensors for traffic detection and an Arduino Mega 2560 microcontroller programmed in Embedded C through the Arduino IDE. Sensors are strategically placed on each road to monitor traffic flow in real time. Based on sensor input, the controller adjusts signal priorities to optimize vehicle movement across all directions. While ultrasonic sensors are effective under normal lighting conditions, future improvements could involve integrating more advanced sensing technologies for enhanced accuracy.**

Keywords — Traffic congestion, Ultrasonic sensors, Arduino Mega 2560, Density-based signal system, Embedded C, Real-time traffic management, Smart transportation, Intelligent transport system (ITS)

I. INTRODUCTION

In today's rapidly urbanizing world, traffic congestion has emerged as a critical challenge in metropolitan areas. With the continuous increase in the number of vehicles and limited expansion of road infrastructure, traditional traffic control methods have become inadequate. Traffic congestion results in longer travel times, increased fuel consumption, elevated stress levels among commuters, and adverse environmental impacts due to vehicular emissions.

Conventional traffic light systems operate on pre-defined, fixed-time intervals, regardless of actual vehicle flow on roads. This leads to inefficient traffic management, especially at multi-way intersections, where one road may have heavy traffic while others remain relatively free. As a result, vehicles on less crowded roads are forced to wait unnecessarily, which further contributes to traffic jams and commuter dissatisfaction.

To address these issues, automated traffic control systems have gained significant attention in recent years. Among these, density-based traffic control systems are particularly effective. These systems monitor the number of vehicles on each road in real-time using sensors and dynamically adjust signal timings to optimize the flow of traffic. The integration of IoT (Internet of Things) technologies with embedded systems allows such traffic control mechanisms to be both cost-effective and scalable.

In this paper, we propose and implement a density-based automatic traffic control system using ultrasonic sensors and an Arduino Mega 2560 microcontroller. The system continuously senses the traffic density on all roads at an intersection and allocates green signal durations accordingly. This approach enhances traffic management by reducing idle time, improving travel efficiency, and ensuring smoother traffic flow.

II. LITERATURE REVIEW

Qasim, K. R., Naser, N. M., and Jabur, A. J. (2024), in their paper "An IoT-Enhanced Traffic Light Control System with Arduino and IR Sensors for Optimized Traffic Patterns" published in Future Internet (16(10):377, DOI: 10.3390/fi16100377), designed a traffic light system using Arduino Mega and infrared (LDR+laser) beam sensors to count vehicles. Data was transmitted to the ThingSpeak cloud platform via an ESP8266 Wi-Fi module. By applying Particle Swarm Optimization (PSO) and Grey Wolf Optimization (GWO), they dynamically adapted green-light durations to traffic density. The system significantly improved flow efficiency, but it was limited to single-intersection deployments and did not incorporate emergency vehicle detection or camera-based recognition.

Khan, A., Khattak, K. S., Khan, Z. H., Gulliver, T. A., and Abdullah (2023), in "Edge Computing for Effective and Efficient Traffic Characterization" (Sensors, 23(23):9385, DOI: 10.3390/s23239385), implemented a Raspberry Pi with Intel Neural Compute Stick and MobileNet-SSD for real-time vehicle detection using computer vision. Their system counted and classified vehicles and transmitted metrics (speed, volume, direction) to ThingSpeak cloud using edge processing, reducing data bandwidth and improving real-time responsiveness. However, the system focused more on traffic characterization than adaptive signal control.

Ashkanani, M., AlAjmi, A., Alhayyan, A., Esmael, Z., AlBedaiwi, M., and Nadeem, M. (2025) proposed a self-adaptive camera-based traffic system in “A Self-Adaptive Traffic Signal System Integrating Real-Time Vehicle Detection and License Plate Recognition for Enhanced Traffic Management” published in Inventions (10(1):14, DOI: 10.3390/inventions10010014). It uses YOLOv11 for vehicle detection and Tesseract OCR for number plate recognition. A web-based digital twin dashboard visualizes real-time traffic and signal changes. The system demonstrated a 95.1% detection accuracy and was highly effective in congestion management, though it depends on high-performance hardware and is not yet suitable for low-cost IoT edge devices.

Khan, A., Ullah, F., Kaleem, Z., Rahman, S. U., Anwar, H., and Cho, Y.-Z. (2018), in “EVP-STC: Emergency Vehicle Priority and Self-Organising Traffic Control at Intersections Using Internet-of-Things Platform” (IEEE Access, DOI: 10.1109/ACCESS.2018.2879644), developed an IoT-enabled system for emergency vehicle prioritization. It uses ZigBee-based traffic sensors to detect vehicle density and GPS modules in emergency vehicles to calculate proximity. The controller adjusts signal timings dynamically, giving priority to emergency vehicles while managing flow. The system showed significant improvements in delay reduction but was dependent on reliable GPS availability and did not detail real-world hardware deployment.

Nellore, K. and Hancke, G. P. (2016), in “Traffic Management for Emergency Vehicle Priority Based on Visual Sensing” published in Sensors (16(11):1892, DOI: 10.3390/s16111892), proposed a vision-based emergency vehicle priority algorithm using distance estimation from video frames. Different distance metrics (Euclidean, Manhattan, Canberra) were tested, and a priority-enhanced MAC (PE-MAC) wireless protocol was developed for timely communication. While the system reduced emergency vehicle response time, it required uninterrupted camera visibility and was tested in a controlled simulation, without edge or cloud IoT integration.

Rosayyan, P., Paul, J., Subramaniam, S., and Ganesan, S. I. (2023), in “An Optimal Control Strategy for Emergency Vehicle Priority System in Smart Cities Using Edge Computing and IoT Sensors” (Measurement: Sensors, vol. 26, Article 100697), presented a GPS-based edge-computing framework that detects emergency vehicles and calculates optimal green-light timing. The system achieved a 73.2% reduction in wait time and sub-100ms latency in field testing. Although practical, the system focused on single-junction scenarios and did not address city-scale coordination or data visualization.

Summary of Key Findings and Gaps: The reviewed literature confirms the growing maturity of IoT-based traffic control systems. Qasim et al. (2024) and Khan et al. (2023) demonstrate effective use of IR sensors and edge computing for traffic monitoring. Ashkanani et al. (2025) prove the power of deep learning for real-time signal adjustment and license plate capture. Emergency vehicle priority is explored by Khan et al. (2018), Nellore and Hancke (2016), and Rosayyan et al. (2023), showing positive results through IoT sensors and visual detection. However, most works remain limited in scope—either to single intersections or without integration of multiple technologies like cloud dashboards, low-cost edge hardware, and emergency detection in one system. The proposed project addresses this gap by combining IR sensors, microcontroller-based hardware, real-time processing, and future extensions toward emergency vehicle handling and mobile dashboard integration.

III. EXISTING SYSTEM

Nowadays, traffic lights are usually placed at intersections and operate using a fixed time cycle, switching signals in a set order regardless of the actual traffic conditions. This traditional method follows a specific pattern where each lane gets a green signal for a fixed duration, even if there are no vehicles waiting on that lane. This results in unwanted congestion, especially during peak hours, and causes delays and frustration for road users.

The current traffic systems do not perform intelligent analysis to decide which signal should be green based on vehicle presence or density. Most traffic signals today are programmed when the junction is first designed, and their timings rarely change, even when traffic patterns shift due to urban growth, road construction, or special events. These systems may work well when traffic is evenly distributed, but they become very inefficient during rush hours or emergencies.

Simple problems such as a road with high vehicle load or the need for priority access for emergency vehicles are not handled by these traditional systems. This leads to vehicles waiting unnecessarily, wasting time and fuel, and increasing pollution levels. In fact, these systems lack real-time sensing and decision-making capability, which is essential in today’s fast-moving urban traffic environment.

To solve these issues, smarter systems are needed that can dynamically adjust signals based on traffic conditions and reduce delays, especially on the more congested roads.

IV. PROPOSED SYSTEM

The proposed system is a density-based automatic traffic control mechanism designed to overcome the inefficiencies of conventional fixed-time traffic lights. In this model, traffic signal durations are dynamically adjusted based on the real-time traffic density present at each approach to a road junction. This enables an intelligent and adaptive method of traffic management, ensuring better vehicle flow, especially during peak hours.

To implement this system, ultrasonic sensors are installed at fixed distances from the signal poles on each road connected to the intersection. These sensors emit ultrasonic waves and measure the time it takes for the waves to reflect back after hitting a vehicle. Using this method, the number of vehicles waiting in each lane can be accurately calculated.

The sensor outputs are connected to an Arduino Mega 2560 microcontroller, which processes the incoming data from all lanes. A program written in Embedded C using the Arduino IDE is deployed to analyze the vehicle counts from each direction. Based on the traffic density data, the microcontroller assigns dynamic green light durations: the lane with the highest vehicle density receives the maximum time, while lanes with fewer vehicles are allocated proportionally shorter durations. If no vehicles are detected in a particular lane, the system either skips the green signal or activates a yellow light to indicate caution.

This traffic control process runs continuously in a cyclic loop. After completing one full signal cycle, the system re-evaluates the traffic density using the sensors and updates the green signal durations accordingly. This ensures that traffic control remains real-time and adaptive, adjusting automatically to changing traffic conditions throughout the day.

The key benefits of the proposed system include reduced idle waiting time, lower fuel consumption, decreased air pollution, improved road efficiency, and higher vehicle throughput at intersections.

The system also offers promising opportunities for future scalability. It can be upgraded to include emergency vehicle detection, pedestrian crossing integration, and wireless communication with smart traffic networks and IoT frameworks.

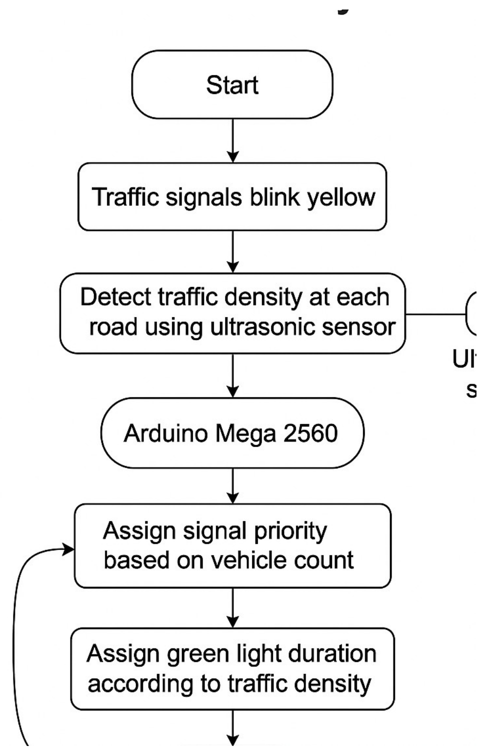
By leveraging cost-effective components like ultrasonic sensors and Arduino-based microcontrollers, the proposed system offers a low-cost, scalable, and efficient solution to modern traffic problems. It is particularly suited for developing countries, where infrastructure upgrades are often constrained by limited budgets.

V. METHODOLOGY

As shown in the methodology diagram, the process begins with powering on the system, which initializes all the components including the traffic signals, sensors, and the Arduino controller. In the initial state, all traffic signals blink yellow, indicating that the system is in a ready and functional state.

The next step is to monitor the traffic density on each road connected to the intersection. Here, traffic density refers to the number of vehicles present on a road at a particular time. This measurement is performed using ultrasonic sensors, which are installed at fixed points on every lane approaching the junction.

These sensors operate by emitting ultrasonic pulses and measuring the time it takes for the signal to return after reflecting off a vehicle. The number of such pulses cut by vehicles is counted and passed to the Arduino Mega 2560, which processes this data in real time.



Once the traffic count from all roads is obtained, the microcontroller analyzes the vehicle density on each side. Based on the highest density detected, the Arduino decides which road should be prioritized and how long the green signal should remain active for that road. The system uses predefined time delays for different density levels—for example, a higher number of vehicles results in a longer green signal.

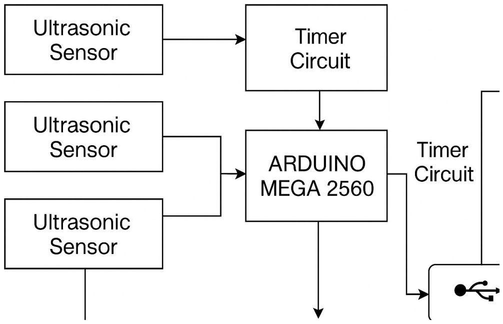
Simultaneously, roads with less traffic are assigned shorter time delays, and if no vehicles are detected, the system either skips the green light or keeps the yellow light blinking as an indication of system readiness.

After allowing each road to pass traffic based on its priority, the system returns to the monitoring phase. The Arduino rechecks the current vehicle density using the sensors and repeats the same decision-making process in a continuous cycle.

The important point to note in this process is that signal timing is never fixed; it adapts dynamically to real-time traffic conditions. This not only reduces waiting time but also ensures efficient traffic flow, less fuel consumption, and minimum congestion.

VI. BLOCK DIAGRAM

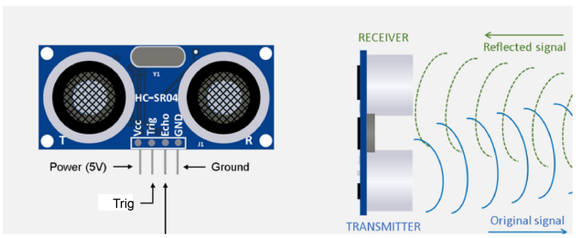
The block diagram of the Automatic Traffic Control System consists of several interconnected hardware and software components that work together to sense traffic density and control traffic signals accordingly. The major components include ultrasonic sensors, Arduino Mega 2560, timer circuits, and the Arduino IDE environment for programming in Embedded C.



Below is a detailed description of each key module

Sensors

Sensors are crucial in embedded systems, especially for real-time monitoring and control applications. In this project, ultrasonic sensors (HC-SR04) are used to detect the presence and count of vehicles approaching an intersection.



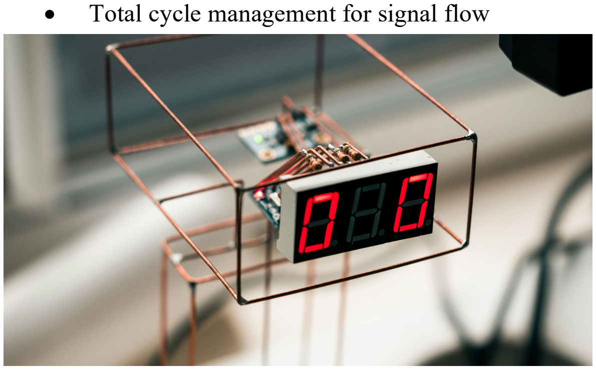
These sensors work by emitting an ultrasonic wave, which reflects back when it hits a vehicle. The sensor calculates the distance based on the time it takes for the echo to return:
Distance = Speed × Time
Since the speed of sound in air is approximately 330 m/s, the sensor calculates the distance accurately. Each time a vehicle crosses the sensing range, the data is sent to the Arduino Mega 2560, which processes this information to estimate traffic density in each lane.
The key benefits of using ultrasonic sensors include:

- Non-contact sensing
- Cost-efficiency
- Real-time responsiveness
- Ease of integration with microcontrollers

Timer

A timer is a specialized clock used for tracking time intervals. In embedded systems, timers are either hardware-based or software-based. In this project, timers are used to control the duration of traffic signals based on sensor input.
Timers help assign:

- Delay for green lights according to vehicle count
- Pause between signal transitions
- Total cycle management for signal flow

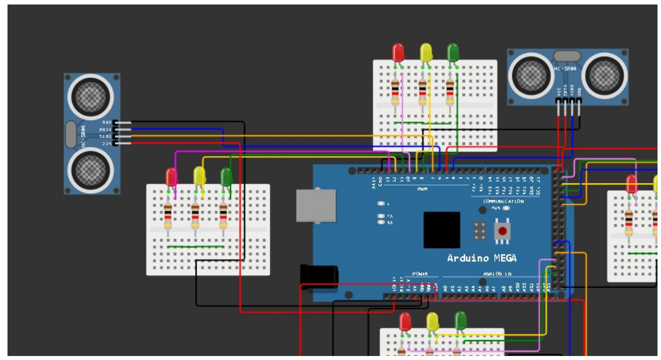


A timer counts upwards (like a stopwatch) or downwards (like a countdown) to determine time-based actions. The microcontroller uses these timer values to adjust light signals dynamically.

Arduino Mega 2560

The Arduino Mega 2560 is the brain of the system. It is an open-source microcontroller board based on the ATmega2560 processor. It features:

- 54 digital I/O pins
- 16 analog input pins
- 4 UARTs (hardware serial ports)
- Ample flash memory and RAM for complex programs

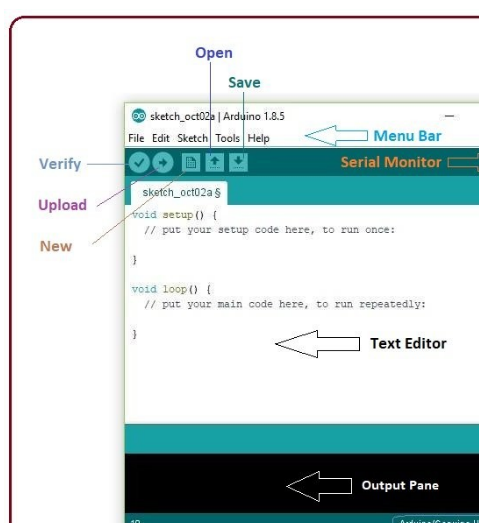


This board reads sensor inputs, executes control logic written in Embedded C, and controls traffic light outputs based on vehicle density.
It can be powered through a USB cable, power jack, or Vin pin. The board uses a bootloader that supports quick and easy uploading of programs via USB without the need for an external programmer.

Arduino IDE

The Arduino IDE (Integrated Development Environment) is the official programming environment used to write, compile, and upload code to Arduino boards. It supports C and C++ languages and is compatible with all Arduino modules.
This software makes it easy to program the logic for:

- Reading sensor input
- Running decision-making algorithms
- Controlling LED signal output
- Managing time delays



The programs written in the IDE are called sketches, and these are uploaded to the board using a USB cable.

Embedded C Programming

Embedded C is the backbone of the software logic that drives the entire system. It is widely used in embedded applications due to its portability, efficiency, and hardware-level control.

Earlier, embedded systems were programmed using assembly language, but C became more popular because:

- It is easier to read and maintain
- It supports modularity and portability
- It integrates well with hardware functions like I/O handling, timers, and interrupts

In this project, Embedded C is used to write:

- The sensor reading logic
- Decision-making for assigning traffic signal durations
- Control sequences for LED lights representing signals

These programs interface directly with the microcontroller's hardware architecture to monitor traffic and control signals dynamically.

```
#include <TimerOne.h>
int signal1[] = {23, 25, 27};
int signal2[] = {46, 48, 50};
int signal3[] = {13, 12, 11};
int signal4[] = {10, 9, 8};
int redDelay = 5000;
int yellowDelay = 2000;
volatile int triggerpin1 = 31;
volatile int echopin1 = 29;
volatile int triggerpin2 = 44;
volatile int echopin2 = 42;
volatile int triggerpin3 = 7;
volatile int echopin3 = 6;
volatile int triggerpin4 = 5;
volatile int echopin4 = 4;
volatile long time;                    // Variable for storing the time traveled
volatile int S1, S2, S3, S4;           // Variables for storing the distance covered
int t = 5;  // distance under which it will look for vehicles.
void setup(){
  Serial.begin(115200);
  Timer1.initialize(100000);  //Begin using the timer. This function must be called first. "microseconds" is the period of time the timer takes.
  Timer1.attachInterrupt(softInterr); //Run a function each time the timer period finishes.
```

Each microcontroller-based system operates using embedded software, which controls the system's behavior. In this project, Embedded C is used to program the Arduino Mega 2560, enabling it to process sensor data and control traffic signals dynamically.

Embedded C is widely preferred for microcontroller programming due to its efficiency, portability, and hardware-level control. Earlier systems used assembly language, which lacked portability. This was overcome with high-level languages like C, which remains the standard in embedded systems today.

To write effective Embedded C programs, developers must understand the microcontroller's architecture, including:

- I/O control
- Timers
- Interrupts
- Serial communication

In this system, Embedded C is used to:

- Read inputs from ultrasonic sensors
- Analyze traffic density
- Assign signal priority and duration
- Control traffic lights in a cyclic and adaptive loop

Programming is done via the Arduino IDE, making it easy to compile and upload code to the board for real-time traffic management.

VII. RESULTS

The Automatic Traffic Control System was successfully implemented using ultrasonic sensors and an Arduino Mega 2560 microcontroller. The prototype was tested on a model representing a four-way junction with varying levels of traffic on each lane.

During testing, the system was able to:

- Accurately detect the number of vehicles on each lane using ultrasonic sensors.
- Dynamically assign green signal durations based on the real-time traffic density detected.
- Skip or minimize green time for lanes with no vehicles, activating a yellow signal instead to indicate system readiness without unnecessary delay.
- Operate in a continuous sensing and control loop, ensuring that the system refreshed traffic data after every signal cycle and adjusted signal priorities accordingly.

The Arduino Mega 2560 efficiently processed real-time sensor inputs and executed decision-making algorithms with minimal response time. The traffic signals responded appropriately to the programmed logic and operated without glitches.

Overall, the results confirmed that the system:

- Reduces waiting time at intersections.
- Improves traffic flow efficiency.
- Minimizes fuel consumption and vehicle idling time.
- Operates reliably in real-time scenarios.

This performance demonstrates that the proposed system is a viable and intelligent alternative to conventional fixed-time traffic signals, particularly suitable for urban and semi-urban traffic conditions.

VIII. CONCLUSION

The Automatic Traffic Control System developed in this project offers a practical and intelligent solution to the increasing traffic congestion in urban areas. Traditional fixed-time traffic signal systems lack adaptability and do not respond to real-time conditions, often resulting in unnecessary delays and inefficient traffic movement. By using ultrasonic sensors to detect vehicle presence and Arduino Mega 2560 to process data and manage signal timings dynamically, this system ensures that traffic flow is optimized based on actual road usage. The implementation showed that the system could effectively prioritize traffic lanes with higher density while minimizing waiting time and fuel wastage on roads with little to no traffic. Furthermore, the use of affordable and readily available components makes the system cost-effective and suitable for deployment in both urban and semi-urban settings, particularly in developing countries.

The intelligent control mechanism significantly reduces idle vehicle time, leading to lower emissions and improved fuel efficiency. This contributes not only to better traffic regulation but also to environmental sustainability. By integrating real-time data collection, decision-making algorithms, and hardware control, the system creates a responsive and adaptive traffic environment. The success of this prototype confirms the potential for broader implementation and further technological enhancement. Overall, the project has met its intended objectives by developing a smart, efficient, and scalable traffic control system capable of adapting to varying traffic conditions and providing a strong foundation for future improvements in intelligent transportation systems.

IX. FUTURE SCOPE

The future scope of your Automatic Traffic Control System project is quite vast and can impact multiple areas in smart city development, IoT, and urban planning. Here are a few potential directions for your project:

1. Integration with Smart City Infrastructure: As cities move toward becoming smart cities, integrating your traffic control system with other city-wide systems such as emergency services, weather forecasting, or urban transportation (e.g., buses, taxis, and ride-sharing) can improve traffic flow, reduce congestion, and enhance safety.
2. AI and Machine Learning for Predictive Analytics: Incorporating AI and machine learning algorithms to predict traffic patterns, accidents, or road closures can improve decision-making. These technologies can learn from historical data to optimize traffic light timings and suggest alternate routes in real time.
3. Vehicle-to-Infrastructure (V2I) Communication: Developing communication between vehicles and traffic systems (V2I) will enable a more responsive system that can adapt to live road conditions, such as changing traffic loads, accidents, or construction zones. This can further reduce congestion and accidents.
4. Environmental Impact Monitoring: You can add features to monitor air quality or carbon emissions. By optimizing traffic flow, your system could contribute to reducing pollution and improving urban air quality.
5. Autonomous Vehicle Integration: With the rise of autonomous vehicles, traffic control systems will need to accommodate these vehicles. Your project could be expanded to interface with autonomous vehicle systems for smoother integration into existing traffic flows.
6. Mobile App Integration: Developing a mobile app that communicates with the traffic control system to give drivers real-time traffic updates, recommended routes, and parking availability could enhance the user experience and improve the system's effectiveness.
7. Scalability to Other Regions: Your system could be scaled to other regions or countries, with slight modifications for local traffic regulations and infrastructure. This would give your project a global impact.
8. 5G Networks for Real-time Data Processing: With the rollout of 5G, you could incorporate faster, real-time data processing to make more immediate traffic flow adjustments based on live traffic conditions.
9. Safety Features: Future iterations could include features like accident detection, automatic response triggering (alerting police or emergency services), and pedestrian detection to make intersections safer.
10. Blockchain for Secure Data Sharing: As traffic systems collect a lot of data, ensuring that this data is securely shared and managed will be crucial. Blockchain could be employed to provide a transparent, decentralized, and tamper-proof system for managing data.

X. REFERENCES

[1] Moraga, Á., de Curtò, J., de Zarzà, I., Calafate, C.T., "AI-Driven UAV and IoT Traffic Optimization: Large Language Models for Congestion and Emission Reduction in Smart Cities", *Drones*, 2025.

[2] Kheder, M.Q., Mohammed, A.A., "Real-time traffic monitoring system using IoT-aided robotics and deep learning techniques", *Kuwait Journal of Science*, 2023.

[3] Bhura, A., Sahoo, P., Kumar, A., Choudhary, U., "IoT-Based Vehicle Class Detection for Smart Traffic Control", *Lecture Notes in Networks and Systems*, Springer, 2024.

[4] Balasubramanian, S.B., Balaji, P., Munshi, A., Almukadi, W., Prabhu, T.N., Venkatachalam, K., Abouhawwash, M., "Machine learning based IoT system for secure traffic management and accident detection in smart cities", *PeerJ Computer Science*, 2023.

[5] Rabby, M.K.M., Shawkat, S.A., Alam, M.S., "A Review of IoT Application in a Smart Traffic Management System", *2019 5th International Conference on Advances in Electrical Engineering (ICAEE)*, IEEE, 2019.

[6] Mohd Ali, A., Al-qerem, A., Hassan, M.R., Abu-Khadrah, A., Jarrah, M., "Traffic Monitoring and Control System for Smart City Pollution Regulation using IoT and Correlated Capsule Networks", *IEEE Access*, 2025.

[7] Jha, A., Agrawal, P., Kaur, R., Agrawal, A., Bhattacharya, M., "IoT Based Smart Traffic Controller System", *2023 Second International Conference on Informatics (ICI)*, IEEE, 2023.

[8] Sadhukhan, P., Gazi, F., "An IoT Based Intelligent Traffic Congestion Control System for Road Crossings", *2018 International Conference on Communication, Computing and Internet of Things (IC3IoT)*, IEEE, 2018.

[9] Pandurangan, R., Vadlamudi, S.G., Kumar, G.V., Bhavani, K., "IOT Based Traffic Controlling System", *2022 International Conference on Smart Generation, Communication and Networking (SMART GENCON)*, IEEE, 2022.

[10] U.S. Government Accountability Office, "Intelligent Transportation Systems: Benefits Related to Traffic Congestion and Safety Can Be Limited by Various Factors", https://www.gao.gov/assets/gao-23-105740.pdf, 2023.

[11] Syed, A.S., Sierra-Sosa, D., Kumar, A., Elmaghraby, A., "IoT in Smart Cities: A Survey of Technologies, Practices and Challenges", *Applied System Innovation*, 2021.